TEC-0082

The UMass RADIUS Project - Year 2

Robert Collins
Allen Hanson
Edward Riseman

University of Massachusetts
Department of Computer Science
Box 34610
Amherst, MA 01003-4610

October 1996

Approved for public release; distribution is unlimited.

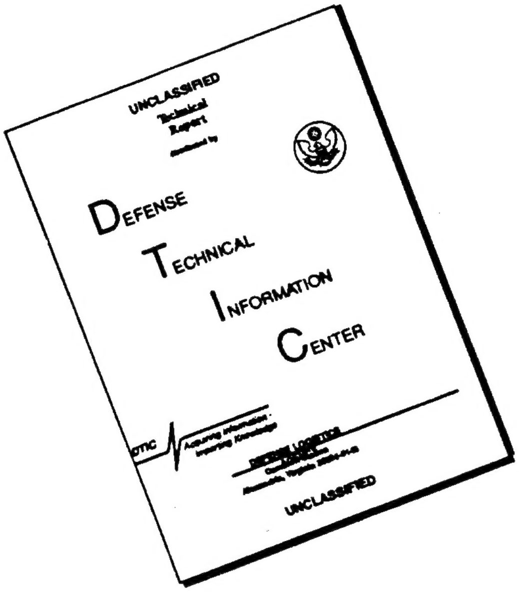
Prepared for:
Defense Advanced Research Projects Agency
3701 North Fairfax Drive
Arlington, VA 22203-1714

Monitored by:
U.S. Army Corps of Engineers
Topographic Engineering Center
7701 Telegraph Road
Alexandria, Virginia 22315-3864

US Army Corps
of Engineers
Topographic
Engineering Center

T
E
C

U.S. ARMY TOPOGRAPHIC ENGINEERING CENTER
TEC



REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** October 1996	**3. REPORT TYPE AND DATES COVERED** Technical October 1993-October 1994

4. TITLE AND SUBTITLE

The UMass RADIUS Project - Year 2

5. FUNDING NUMBERS

DACA76-92-C-0041

6. AUTHOR(S)

Robert Collins, Allen Hanson, Edward Riseman

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

University of Massachusetts
Department of Computer Science
Box 34610
Amherst, MA 01003-4610

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Defense Advanced Research
Projects Agency
3701 N. Fairfax Drive
Arlington, VA 22203-1714

U.S. Army Topograpphic
Engineering Center
7701 Telegraph Road
Alexandria, VA 22315-3864

19. SPONSORING / MONITORING AGENCY REPORT NUMBER

TEC-0082

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

A set of image understanding (IU) algorithms for automated site model acquisition and extension are being developed at the University of Massachusetts. In 1995, a building extraction system was implemented to acquire flat roofed, rectilinear building models from multiple, monocular images. This system hypothesizes building rooftops from a single image, then searches for supporting evidence in other views and determines the precise 3-D shape and location of each building via multi-image triangulation. Projective mapping of image intensity information onto these polyedral building models results in a realistic site model that can be rendered using virtual "fly-through" graphics. To perform model extension, a prior site model is registered to new images, and building model acquisition procedures are focused on previously unmodeled areas. In an operational scenario, this process would be repeated as new images become available, gradually accumulating evidence over time to make the site model data base more complete and accurate. Model-to-image registration techniques also are presented that can be used to automatically determine model-based local corrections to the resected camera parameters provided with each image.

14. SUBJECT TERMS

Building Extraction, Site Model Acquisition and Extension, Aerial Image Understanding, 3-D Scene Visualization

15. NUMBER OF PAGES

37

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UNLIMITED

Contents

List of Figures

List of Tables

PREFACE

This research was sponsored by the Defense Advanced Research Projects Agency (DARPA) and was monitored by the U.S. Army Topographic Engineering Center (TEC) under Contract DACA76-92-C-0041, titled "The UMass RADIUS Project - Year 2." The DARPA Program Manager is Dr. Oscar Firschein and the TEC Contracting Officer's Representative is Ms. Lauretta Williams.

1 Introduction

The goal of the Research and Development for Image Understanding Systems (RADIUS) project is to develop image understanding (IU) algorithms that support model-based aerial photointerpretation. The Computer Vision Research Laboratory at the University of Massachusetts (UMass) is funded under a three-year contract to develop algorithms to automatically acquire, extend, and refine 3D geometric site models. This report describes research activities performed during the second year of the UMass RADIUS contract, covering the period of October 1993 – October 1994.

Site **model acquisition** involves processing a set of images to detect both man-made and natural features of interest, and to determine their 3D shape and placement in the scene. The models produced have obvious applications in areas such as surveying, surveillance and automated cartography. For example, acquired site models can be used for automated model-to-image registration of new images, allowing the model to be overlaid on the image to aid visual change detection and verification of expected scene features. Two other important site modeling tasks are **model extension** – updating the geometric site model by adding or removing features, and **model refinement** – iteratively refining the shape and placement of features as more views become available. Model extension and refinement are ongoing processes that are repeated whenever new images become available, each updated model becoming the current site model for the next iteration. Thus, over time, the site model has steadily improved to become more complete and more accurate.

In 1995, work at the University of Massachusetts (UMass) focussed on designing and implementing a system for automatically extracting models of buildings from multiple, overlapping images of a site. The system design emphasizes model-directed processing, rigorous camera geometry, and fusion of information across multiple images for increased accuracy and reliability. To maintain a tractable goal for our research efforts, we have chosen initially to focus on a single generic class of buildings, namely flat-roofed, rectilinear structures. The simplest example of this class is a rectangular box-shape; however other examples include L-shapes, U-shapes, and indeed any arbitrary building shape such that pairs of adjacent roof edges are perpendicular and lie in a horizontal plane. The system is designed to operate over multiple images exhibiting a wide variety of viewing angles and sun conditions. The system is designed to perform well at one end of a data-vs-control complexity spectrum, namely a large amount of data and a simple control structure, versus the alternative of using less data but more complicated processing strategies. In particular, while the system can be applied to a single stereo pair, it generally performs better (in terms of number and quality of buildings found) when more images are used.

This document proceeds as follows. Section 2 begins with a specification of general input requirements for the UMass building extraction system. This is followed in Section 3 by a breakdown of the system into its key algorithmic components. Section 4 reports on an experimental case study in site model acquisition, using the RADIUS Model Board 1 data set. Section 5 describes the process of model extension, and provides a demonstration using the partial site model acquired in Section 4. Section 6 presents a brief summary and a set of likely program activities for next year.

2 General System Requirements

The UMass building extraction system was developed on a Sun Sparc 10, using the RADIUS Common Development Environment (RCDE) [8]. The RCDE is a combined Lisp/C++ system that supports the development of image understanding algorithms for constructing and using site models. In particular, the RCDE provides a convenient framework for representing and manipulating images, camera models, object models and terrain models, and for keeping track of their various coordinate systems, inter-object relationships, and transformation/projection equations. The RCDE also provides utilities for interactively developing site models, specifying tie points, and for performing photo-resection.

2.1 Images

Acquisition of a 3D site model requires a set of overlapping images of the site. The UMass system is designed to operate over multiple images, typically five or more, exhibiting a wide variety of viewing angles and sun conditions. The number five is chosen arbitrarily to allow one nadir view plus four oblique views from each of four perpendicular directions (e.g. North, South, East and West). This configuration is not a requirement, however. Indeed, some useful portions of the system require only a single image, namely line segment extraction and building rooftop detection. On the other hand, epipolar rooftop matching and wireframe triangulation require, by definition, at least two images, with robustness and accuracy increasing when more views are available. Once again, the number five has been chosen arbitrarily, and perhaps only three well-chosen images would suffice, but verification of this is a matter for further experimentation.

Although best results require the use of many images with overlapping coverage, the system allows considerable freedom in the choice of images to use. Unlike most other building extraction systems, this system does not currently use shadow information, and works best if used on images with different sun angles, or with no strong shadows at all. Also, the term "epipolar" as used here does not imply that images need to be in scan-line epipolar alignment, as required by many traditional stereo techniques. The term is used instead in its general sense as a set of geometric constraints imposed on potentially corresponding image features by the relative orientation of their respective cameras. The relative orientation of any pair of images is computed from the absolute orientation of each individual image (see Section 2.3).

2.2 Site Coordinate System

Reconstructed building models are represented in a local site coordinate system that must be defined prior to the reconstruction process. The system assumes this is a "local-vertical" Euclidean Coordinate System, that is, a Cartesian X-Y-Z coordinate system with its origin located within or close to the site, and the positive Z-axis pointing upwards (parallel to gravity). The system can be either right-handed or left-handed. Under a local-vertical coordinate system, the Z values of reconstructed points represent their vertical position or "height" in the scene, and X-Y coordinates represent their horizontal location in the site.

2.3 Camera Models

For each image given to the system, the absolute orientation of the camera with respect to the local site coordinate system must be known. This includes both the internal orientation (lens/digitizer parameters) and the external orientation (pose parameters) of the camera. Given the absolute orientation for each image, the system computes all the necessary relative orientation information needed for determining the epipolar geometry between images. Camera models can be specified in two ways. For the **perspective frame camera model,** absolute orientation for each camera is supplied as a 3×4 projective transformation matrix describing (in homogeneous coordinates) how points in the site coordinate system project into points in the image coordinate system. This simple representation makes no distinction between internal and external camera parameters. Translation between "standard" photogrammetric parameterizations (e.g. focal length, principle point coordinates, camera location vector and rotation Euler angles) and the 3×4 matrix representation is provided by the RCDE.

Many aerial photographs, particularly satellite images, are generated by nontraditional imaging systems for which the standard perspective frame camera model is not an adequate description. The **fast block interpolation projection** (FBIP) camera model has been proposed as an alternative description of the imaging process in these situations. The general idea is to break space into "blocks" and then generate local frame camera approximations within each block in such a way that adjacent frame approximations agree at the block boundary, in a manner somewhat analogous to approximating a nonlinear function by a piecewise linear one. This representation easily handles 2D image nonlinearities, such as camera lens distortion, as well as 3D space nonlinearities caused by the refraction of light through layers of the atmosphere.

Integrating the FBIP camera model into image understanding algorithms is potentially difficult, since it violates the fundamental assumption underlying most work with traditional, perspective camera models, namely the assumption that straight lines in the world will appear straight in the image. The FBIP camera model not only raises representational concerns, such as whether the edge of a building in the image can be adequately characterized by a single straight line segment, but also strikes at a deeper level, invalidating such fundamental geometric notions as vanishing points and epipolar geometry. Our interpretation of the FBIP camera model is that it is possible to derive a local 3×4 projective transformation matrix that provides an accurate approximation to the imaging process within a given 3D region of interest spanning the spatial extents of a single building.

2.4 Digital Terrain Map

Currently, the UMass system explicitly reconstructs only the rooftops of building structures, and relies on vertical extrusion to form a volumetric 3D wireframe model of the whole building. In other words, perpendiculars are dropped from each corner of the reconstructed building rooftop down to the ground, and connected by a building base formed as a vertical translation of a copy of the roof polygon. The extrusion process relies on knowing the local terrain, namely the ground height (Z value) at each location in the scene. We assume this information is represented as an array of elevations, or in the special case of flat ground planes as a horizontal plane equation $Z = z_0$. Representation of digital terrain maps in either format, along with their use in providing a basic ground level for vertical extrusion, is supported by the RCDE. Future versions of the system will use digital terrain maps automatically extracted from stereo image pairs (nadir or oblique) by a correlation-based terrain reconstruction system developed recently at UMass.

2.5 Other Required Parameters

In addition to the general information described above, a few miscellaneous parameters and thresholds are required to be supplied by the user before the system can be run. The most important of these are:

- **max-building-height** – the maximum possible height of any building that will be included in the site model. This threshold is used to limit the extent of epipolar search regions. The lower this threshold can be, the smaller the search area for rooftop feature matches will be, leading to faster searches with higher likelihood of finding the correct matches.

- **min-building-width** – the minimum horizontal extent (width or length) of any building that will be included in the site model. This is, loosely speaking, a way of specifying the desired "resolution" of the resulting site model, since any buildings having horizontal edges shorter than this threshold will probably not be found. Setting this value to a relatively long length essentially ensures that only large buildings in the site will be modeled.

3 Algorithmic Building Blocks

The UMass building extraction system currently follows a simple processing strategy. To acquire a new site model, an automated building detector is run on one image to hypothesize potential building rooftops. Supporting evidence is located in other images via epipolar line segment matching, and the precise 3D shape and location of each building is determined by multi-image triangulation and extrusion. Image intensity information is backprojected onto each face of these polyhedral building models to facilitate realistic rendering from new views.

This section describes the key algorithms that together comprise the UMass building extraction system. These algorithms are: line segment extraction, building rooftop detection, epipolar rooftop matching, multi-image wireframe triangulation, and projective intensity mapping. Line segment extraction and building rooftop detection are illustrated with sample results from two sites, the Schenectady County Air National Guard base (Figure 1), and Radius Model Board 1 (Figure 2).

3.1 Line Segment Extraction

To help bridge the huge representational gap between pixels and site models, a straight line feature extraction routine is applied to produce a set of symbolic line segments, representing geometric image features of potential interest, such as building roof edges. We use the Boldt algorithm for extracting line segments [3]. At the heart of the Boldt algorithm is a hierarchical grouping system inspired by the Gestalt laws of perceptual organization. Zero-crossings of the Laplacian of the intensity image provide an initial set of local intensity edges. Hierarchical grouping then proceeds iteratively; at each iteration edge pairs are linked and replaced by a single longer edge if their end points are close and their orientation and contrast (difference in average intensity level across the line) values are similar. Each iteration results in a set of increasingly longer line segments. The final set of line segment features (Figures 3 and 4) can be filtered according to length and contrast values supplied by the user.

Although the Boldt algorithm does not rely on any particular camera model, the utility of extracting straight lines as a relevant representation of image/scene structure is based on the assumption that straight lines in the world (such as building edges) will appear reasonably straight in the image. To the extent that this assumption remains true at the scale of the objects being considered, such as over a region of the image containing a single building, then straight line extraction remains a viable feature detection method. However, very long lines spanning a significant extent of the image, such as the edges of airport runways, may become fragmented depending on the amount of curvature introduced into the image by nonlinearities in the imaging process.

3.2 Building rooftop detection

The goal of automated building detection is to roughly delineate building boundaries that will later be verified in other images by epipolar feature matching and triangulated to create 3D geometric building models. The UMass building detection algorithm [6] is based on perceptual grouping of line segments into image polygons corresponding to the boundaries of flat, rectilinear rooftops in the scene. Perceptual organization is a powerful method for locating and extracting scene structure. The rooftop extraction algorithm proceeds in three steps; low-level feature extraction, collated feature detection, and hypothesis arbitration. Each module generates features that are used during the next phase and that interact with

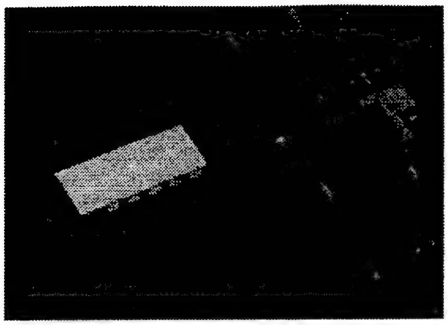


Figure 1: Schenectady subimage.

Figure 2: Model Board 1 subimage.

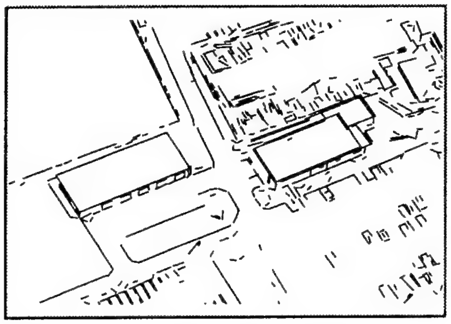


Figure 3: Boldt lines for Figure 1.

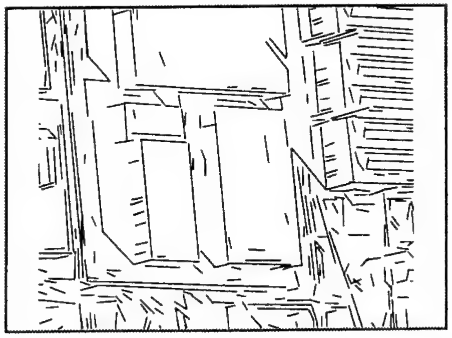


Figure 4: Boldt lines for Figure 2.

lower level modules through top-down feature extraction.

Low-level features in this system are straight line segments and corners. The domain assumption of flat-roofed rectilinear structures implies that rooftop polygons will be produced by flat horizontal surfaces with orthogonal corners. Orthogonal corners in the world are not necessarily orthogonal in the image, however. To determine a set of relevant corner hypotheses, pairs of line segments with spatially proximate endpoints are grouped together into candidate image corner features. Each potential image corner is then backprojected into a nominal Z-plane in the scene, and that hypothetical *scene corner* is tested for orthogonality.

Mid-level collated features are sequences of perceptually grouped corners and lines that form a chain (Figures 5 and 6). A valid chain group must contain an alternation of corners and lines, and can be of any length. Chains are a generalization of the collated features in earlier work [5] and allow final polygons of arbitrary rectilinear shape to be constructed from low-level features. Collated feature chains are represented by paths in a feature relation graph. Low level features (corners and line segments) are nodes in the graph, and perceptual grouping relations between these features are represented by edges in the graph. Nodes have a certainty measure that represents the confidence of the low-level feature extraction routines; edges are weighted with the certainty of the grouping that the edge represents. A chain of collated features inherits an accumulated certainty measure from all the nodes and edges along its path.

High-level polygon hypothesis extraction proceeds in two steps. First, all possible polygons are computed from the collated features. Then, polygon hypotheses are arbitrated in order to arrive at a final set of non-conflicting, high confidence rooftop polygons (Figures 7

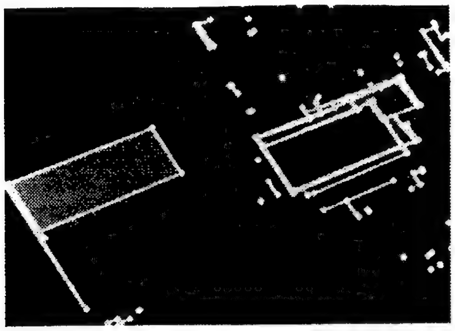


Figure 5: Feature chains for Figure 1.

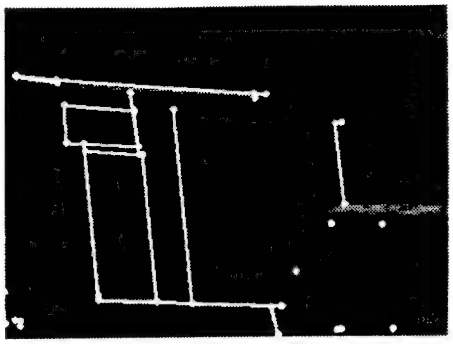


Figure 6: Feature chains for Figure 2.

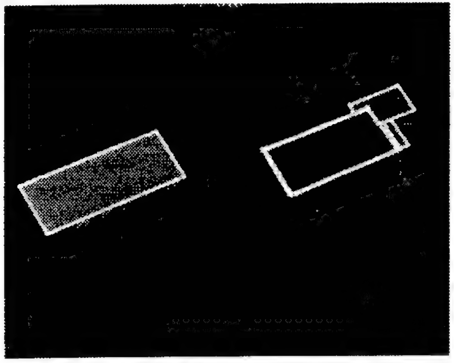


Figure 7: Rooftop hypotheses for Figure 1.

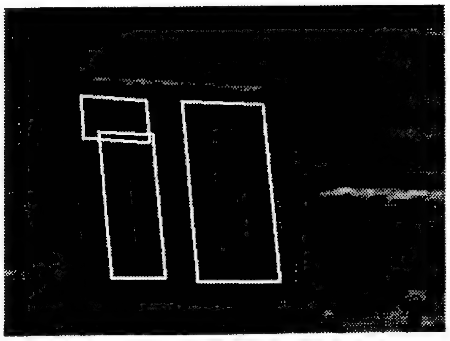


Figure 8: Rooftop hypotheses for Figure 2.

and 8). Polygon hypotheses are simply closed chains, which can be found as cycles in the feature relation graph. All of the cycles in the feature relation graph are searched for in a depth first manner, and stored in a dependency graph where nodes represent complete cycles (rooftop hypotheses). Nodes in the dependency graph contain the certainty of the cycle that the node represents. An edge between two nodes in the dependency graph is created when cycles have low-level features in common. The final set of non-overlapping rooftop polygons is the set of nodes in the dependency graph that are both independent (have no edges in common) and are of maximum certainty. Standard graph-theoretic techniques are employed to discover the maximally-weighted set of independent cycles which is output by the algorithm as a set of independent high confidence rooftop polygons.

While searching for closed cycles, the collated feature detector may be invoked in order to attempt closure of chains that are missing a particular feature (an example occurs in Figure 6). The system then searches for evidence in the image that such a virtual feature can be hypothesized. In this way, the rooftop detection process does not have to rely on the original set of features that were extracted from the image. Rather, as evidence for a polygon accumulates, tailor-made searches for lower level features can be performed. This type of top-down inquiry increases system robustness.

3.3 Epipolar Line Segment Matching

After detecting a potential rooftop in one image, corroborating geometric evidence is sought in other images (often taken from widely different viewpoints) via epipolar feature matching.

The primary difficulty to be overcome during epipolar matching is the resolution of ambiguous potential matches, and this ambiguity is highest when only a single pair of images are used. For example, the epipolar search region for a roof edge match will often contain multiple potentally matching line segments of the appropriate length and orientation, one of which comes from the corresponding roof edge, but the others coming from the base of the building, the shadow edge of the building on the ground, or from roof/base/shadow edges of adjacent buildings (see Figure 9). This situation is exacerbated when the roof edge being searched for happens to be nearly aligned with an epipolar line in the second image. The resolution of this potential ambiguity is the reason that simultaneous processing of multiple images with a variety of viewpoints and sun angles is preferred in the UMass system.

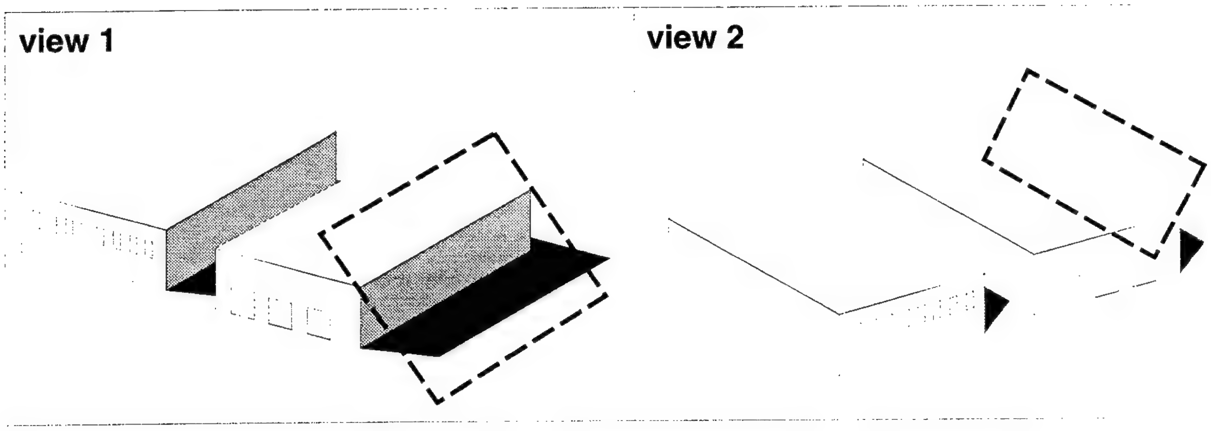


Figure 9: Multiple ambiguous matches can often be resolved by consulting a new view.

We match rooftop polygons by searching for each component line segment separately and then fusing the results. For each polygon segment from one image, an appropriate epipolar search area is formed in each of the other images, based on the known camera geometry and the assumption that the roof is flat. This quadrilateral search area is scanned for possible matching edges, the disparity of each potential match implying a different roof height in the scene. Results from each line search are combined in a 1-dimensional histogram, each potential match voting for a particular roof height. Each vote is weighted by compatibility of the match in terms of expected line segment orientation and length. This allows for correct handling of fragmented line data, for example, since the combined votes of all subpieces of a fragmented line count the same as the vote of a full-sized, unfragmented line. A single global histogram accumulates height votes from multiple images, and for multiple edges in a rooftop polygon. After all votes have been tallied, the histogram bucket containing the most votes yields an estimate of the roof height in the scene and a set of correspondences between rooftop edges and image line segments from multiple views.

3.4 Wireframe triangulation/extrusion

After finding a set of rooftop edge correspondences via epipolar matching, multi-image triangulation is performed to determine the precise size, shape, and position of the roof polygon in the local 3D site coordinate system. A nonlinear estimation algorithm has been developed for simultaneous multi-image, multi-line triangulation of 3D line structures.

Two versions of the triangulation subsystem have been developed. In the first, the parameters estimated for each rooftop edge are the Plücker coordinates of the algebraic 3D line coinciding with the edge. Specific points of interest, like vertices of the rooftop polygon, are computed as the intersections of these infinite algebraic lines. Plücker coordinates are a way of embedding the 4-dimensional manifold of 3D lines into R^6. Although the Plücker representation requires 6 parameters to be estimated for each line rather than 4, it simplifies the representation of geometric constraints between lines. For the generic flat-roofed rectilinear building class being considered here, a set of constraints is specified to ensure that pairs of adjacent lines in a traversal around the polygon are perpendicular, that all lines are coplanar, and that all lines are perpendicular to the Z-axis of the local site coordinate system. An iterative, nonlinear least-squares procedure determines the Plücker coordinates for all lines simultaneously such that all the object-level constraints are satisfied, and an objective "fit" function is minimized that measures how well each projected algebraic line aligns with the 2D image segments that correspond to it.

Although triangulation of line structures via Plücker coordinates is general, in the sense that any set of 3D lines can be represented, we have found this approach to be computationally burdensome and numerically unstable. The reason for this is mainly due to the number of parameters in the representation and the number of constraints that must be imposed to achieve a unique, geometrically accurate solution. In particular, triangulation of a rooftop polygon containing n lines requires $6 \times n$ parameters to represent the Plücker coordinates, plus an addition $2 \times n$ Lagrange multipliers to ensure a unique solution (recall that the dimension of the line manifold is 4, thus $6 - 4 = 2$ additional constraints are required for each line to make the solution vector unique). Further constraints (and thus more Lagrange multiplier parameters) are necessary to impose the required geometric configuration on the lines in the final polygon, namely that all are coplanar and horizontal, and that adjacent pairs are perpendicular.

In response to these computational difficulties, a second version of the triangulation system has been developed using a specialized parameterization for representing flat, rectilinear polygons. The types of line structures that can be triangulated are considerably more restrictive than in the earlier, general version; however, the restrictions mesh well with current system assumptions and result in a much more streamlined optimization problem. Instead of each line being represented separately, a whole rectilinear polygon is parameterized at once, using the variables shown in Figure 10. The horizontal plane containing the polygon is parameterized by a single variable $\boldsymbol{Z}$. The orientation of the rectilinear structure within that plane is represented by a single parameter $\boldsymbol{\theta}$. Finally, each separate line within the polygon is represented by a single value $\boldsymbol{r}_i$ representing the signed perpendicular distance of that line from some nominal point in the plane, usually chosen to be near the center of mass of the polygon being estimated. The representation is simple and compact, and the method of Lagrange multipliers is no longer necessary since the coplanarity and rectilinearity constraints on the polygon's shape are already built in to the representation.

Regardless of which parameterization is chosen, nonlinear estimation algorithms typi-

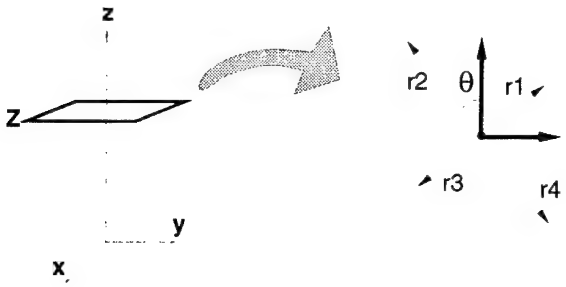


Figure 10: Parameterization of a flat, rectilinear polygon for multi-image triangulation.

cally require an initial estimate that is then iteratively refined. In this system, the original rooftop polygon extracted by the building detector, and the roof height estimate computed by the epipolar matching algorithm, are used to generate an initial, flat, roof polygon. After triangulation, each 3D rooftop polygon is extruded down to the ground, as determined by the digital terrain map for the site (see Section 2.4), to form a volumetric wireframe model.

3.5 Projective intensity mapping

To provide added realism for visual displays, and as a convenient means of storage for later detailed processing of building surface information, mechanisms have been developed for projectively warping image intensities onto polygonal building facets. Planar projective transformations provide a mathematical description of how surface structure from a planar building facet maps into an image. By inverting this transformation using a known building position and camera geometry, intensity information from each image can be backprojected to "paint" the walls and roof of the building model. Since multiple images are used, intensity information from all faces of the building polygon can be recovered, even though they are not all seen in any single image (see Figure 11). The full intensity-mapped site model can then be rendered to predict how the scene will appear from a new view (Figure 12), and on high-end workstations realistic real-time "fly-throughs" can be generated.

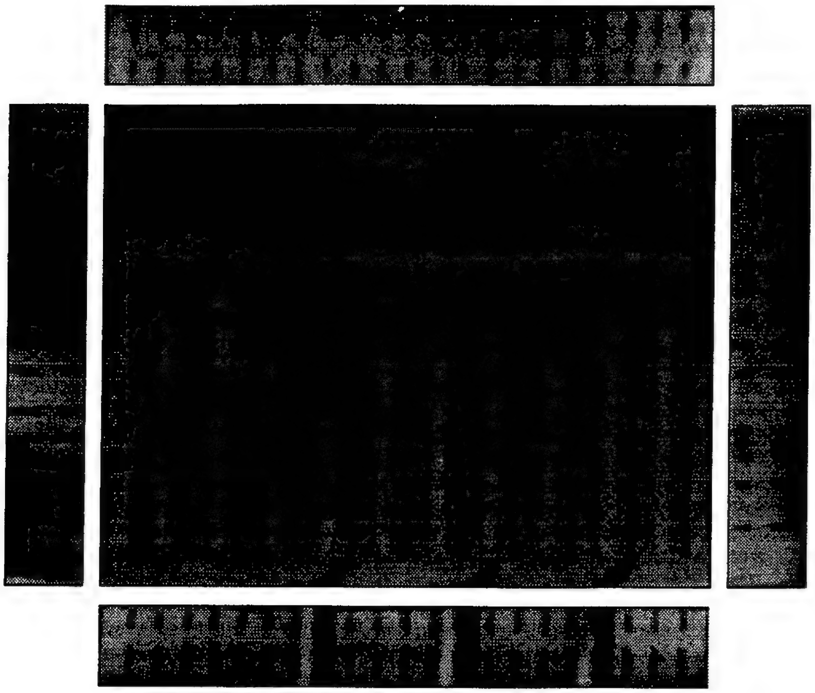


Figure 11: Intensity maps are stored with the planar facets of a building model.

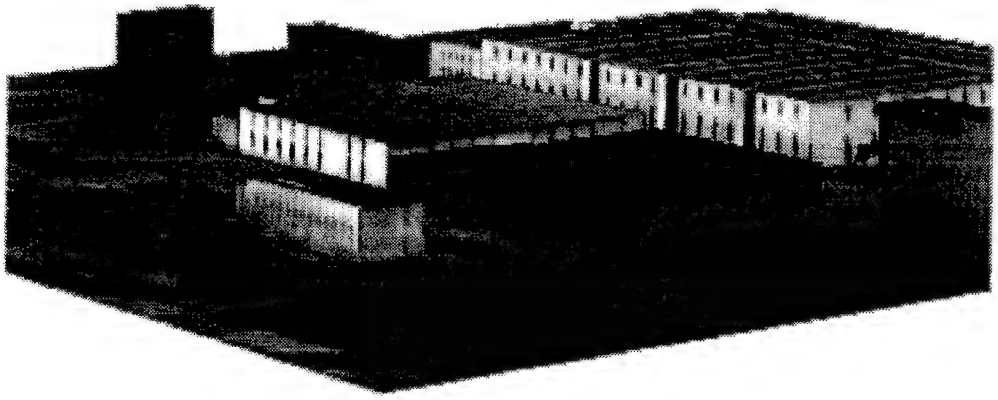


Figure 12: Intensity-mapped site model rendered from a new view.

By storing surface information with the object, intensity mapping provides a convenient storage method for later symbolic extraction of detailed surface structures like windows, doors and roof vents. Furthermore, this subsequent processing becomes greatly simplified. For example, rectangular lattices of windows or roof vents can be searched for in the unwarped intensity maps without complication from the effects of perspective distortion. Secondly, specific surface structure extraction techniques can be applied only where relevant, i.e. window and door extraction can be focused on building wall intensity maps, while roof vent computations are performed only on roofs.

When processing multiple overlapping images, each building facet will often be seen in more than one image, under a variety of viewing angles and illumination conditions. This has led to the development of a systematic mechanism for managing intensity map data, called the Orthographic Facet Library. The orthographic facet library is an indexed data set storing all of the intensity-mapped images of all the polygonal building facets that have been recovered from the site. Usually, a horizontal roof facet appears in all the aerial site images and thus has a complete set of intensity-map versions in the library. Vertical wall facets usually show up only in a subset of the site images, however, so fewer intensity-map versions are available to choose from. Each intensity-map version is tagged with a variety of spatial and photometric indices (e.g. viewing angle, resolution, sun angle) in order to facilitate retrieval and analysis by image understanding algorithms. As intensity-mapped building facets accumulate in the facet image library, knowledge about the site improves; albeit in an implicit, image-based form.

When using the facet library to render a new view of the site, it is necessary to distill the information contained in multiple intensity-mapped versions of each building facet into a single "best" image representation for that facet. Two alternative solutions have been tried so far. The first approach is to use the pixels in the **best representative version** of each facet to paint the given surface. The "goodness" of an image with respect to a particular building facet is based on a heuristic measure that takes into account the camera viewing angle, the sun angle, and the placement and geometry of other buildings in the site, all of which allow the system to compute the size, relative orientation, and photometric contrast of the facet in the image, as well as predict the percentage of the facet covered by shadows or occlusion in that view. The advantage of best version representation is its simplicity, in that only a heuristic function is calculated for each view and no further image processing is needed. The drawback of this method is that sometimes occlusions or shadows appear in every image of a building facet; thus, the representative will have to include those artifacts no matter which image is chosen. The best version representation was used to texture the building facets in Figure 11.

In contrast to the best version approach, the **best representative piece** method takes occlusions and shadows into account. As intensity-map versions are placed in the library, pixels in the facet are partitioned into "pieces" according to whether they are sunlit or in shadow. Pixels that are labeled as occluded areas are discarded and are not considered to be a part of any piece. The idea of the best piece representation is to assign a heuristic value to each piece of an intensity-map version, rather than to the entire version. When rendering a new view, each pixel on a building's surface is backprojected to determine which pieces it is associated with. This set of pieces is ordered according to their heuristic values, and the photometric value for the pixel is selected from the highest-rated piece. Hence, all the pixels in the rendered image are the best ones available. Note, however, that some pixels in the rendered image might not exist in any of the pieces in the library, when they correspond to portions of building that have never been seen in any of the images. These pixels are painted black by default. The best piece representation was used to render the site model in Figure 12.

The best piece representation is a method of data fusion, and compatability problems arise in that different pieces of each building face can appear under different sunlight conditions in different images, and thus different portions of the same building face may be assigned

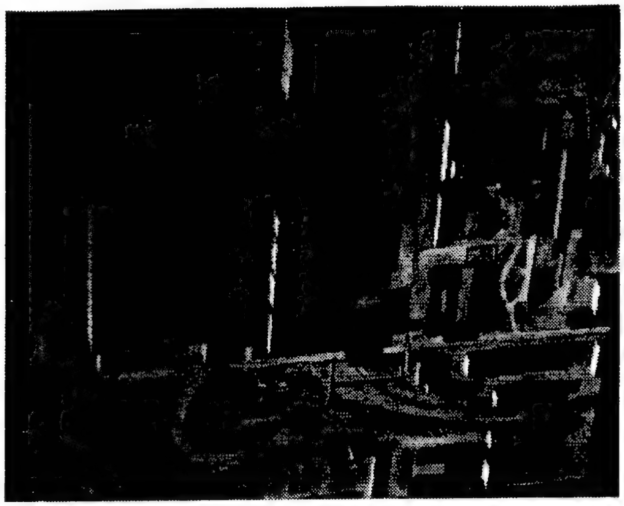


Figure 13: A sample image from Model Board 1

significantly different grey-levels, leading to a patchy appearance. One reasonable way to solve this problem is to make all the versions of the facet "similar" in intensity. Currently, a simple histogram adjustment technique is used to make the intensity distributions of all the pieces associated with a single building face uniform with respect to each other. Currently, the biggest sunlit piece of the facet is chosen as the model piece against which all other pieces are transformed.

4 Model Acquisition Experiment

Previous sections of this report present a system developed at UMass for automated site model acquisition via building extraction. The main algorithmic components of this system are line segment extraction, monocular building rooftop detection, multi-image epipolar rooftop matching, multi-image wireframe triangulation, and projective intensity mapping. This section demonstrates the model acquisition process by way of an experimental case study using images J1–J8 of the RADIUS Model Board 1 data set. In this experiment, 25 building models were generated, covering a large portion of the model board site. The study was conducted in order to exercise and evaluate our current model acquisition system on a realistic task. The goal of this section is to present a fair evaluation of current system performance by showing representative successes, failures, and a quantitative analysis of results.

4.1 Radius Model Board 1

The model acquisition experiment was performed using images J1–J8 from the RADIUS Model Board 1 data set. Figure 13 shows a sample image from the data set. The scene is a 1:500-inch scale model of an industrial site. Ground truth measurements are available for about 110 points scattered throughout the model. The scale model is built on a table top that can be raised and tilted to simulate a variety of camera altitudes and orientations. For model board images J1–J8, the table was set to simulate aerial photographs taken with a ground sample distance of 18 inches, that is, pixels near the center of the image backproject to quadrilaterals on the ground with sides approximately 18 inches long (all measurements will be reported in scaled-up (i.e. ×500) object coordinates). Each image contains approximately 1320×1035 pixels, with about 11 bits of grey level information per pixel. The dimensions of each image vary slightly because the images have been resampled and subjected to unmodeled geometric and photometric distortions that simulate actual operating conditions.

4.2 Camera Resection

Camera resection (calibration) to determine the interior (lens) and absolute (pose) parameters of each camera with respect to the scene is a precursor for many site-modeling tasks, and is essential for accurate 3D triangulation of scene features. The resection process determines both the absolute orientation of an image (the precise geometric relationship between the image and the scene) and the relative orientation between images (that is their relationship relative to each other, which in essence determines their epipolar geometry). Ideally, images to be used for site-modeling purposes would be resected prior to the application of IU algorithms for automated building extraction. Indeed, that is the goal of the ARPA/ORD Model Supported Positioning (MSP) project. The model board images were not supplied with an accurate set of camera parameters, however.

We resected images J1–J8 by directly estimating the 11 free parameters of the 3×4 projective transformation matrix for each image. Matrix elements were computed by setting the lower right-hand element of the projective transformation matrix to 1, then estimating the remaining elements using an iterative least-squares procedure to minimize the sum of squared residual errors between projected 3D ground truth points and their hand-selected 2D image locations. Note that more sophisticated resection procedures are available that take into account prior knowledge of the internal camera parameters and enforce geometric consistency constraints between solutions for multiple images [1]. However, our simple approach was easy to implement and run, and in fact worked well for this set of images because a large number of well-distributed ground control points are available.

Table 1: RMS errors (in pixels) for J1–J8 resections

image number	J1	J2	J3	J4
RMS error	1.95	1.93	2.72	2.38
image number	J5	J6	J7	J8
RMS error	2.25	2.87	2.38	2.04

Table 1 shows the average residual error for the resections we performed. The residual error for each image is in the 2 to 3 pixel range, representing the level of unmodeled geometric distortion present in each image. Since the ground scale distance is 18 inches, this

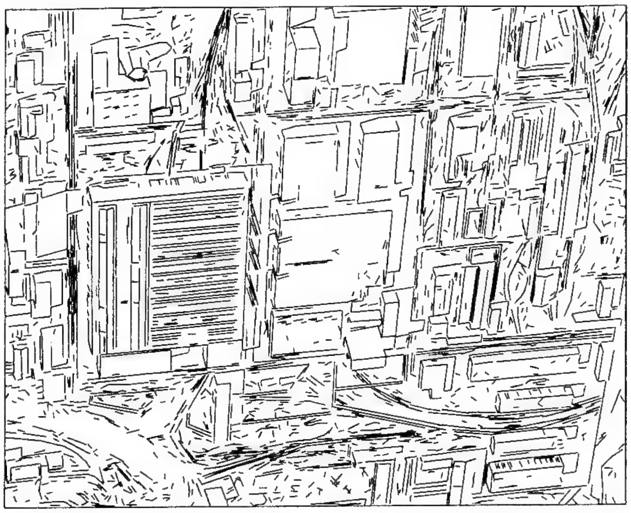


Figure 14: Boldt line segments extracted from Figure 13

corresponds to a backprojection error of roughly 3 to 4.5 feet in object space. This is a significant amount of error, and presents a good test of system robustness.

We note in passing that we also have a model-to-image registration system that automatically determines the correspondence between 3D building wire-frame edges and 2D image line segments while simultaneously solving for a robust estimate of camera pose (see Section 5. This resection method was not appropriate for this experiment, however, since we wanted to simulate an initial model acquisition episode where there were no prior building models.

4.3 Line Segment Extraction

To help bridge the huge representational gap between pixels and site models, feature extraction routines are applied to produce symbolic, geometric representations of potentially important image features. We use the Boldt algorithm for extracting line segments [3]. At the heart of the Boldt algorithm is a hierarchical grouping system inspired by the Gestalt laws of perceptual organization. Zero-crossing points of the Laplacian of the intensity image provide an initial set of local intensity edges. Hierarchical grouping then proceeds iteratively; at each iteration edge pairs are linked and replaced by a single longer edge if their end points are close and their orientation and contrast (difference in average intensity level across the line) values are similar. Each iteration results in a set of increasingly longer line segments. The final set of line segments were filtered to have a length of at least 10 pixels long and a contrast of at least 15 grey levels. This procedure produced roughly 2800 line segments per image. Figure 14 shows a representative set of lines, extracted from the image shown in Figure 13.

4.4 Building Detection

The goal of automated building detection is to roughly delineate building boundaries that will later be verified in other images by epipolar feature matching and triangulated to create 3D geometric building models. The building detector was run on image J3. This happens to be a near-nadir view, but nothing in the code precludes using one of the oblique views instead (see [6]). The roof detector generated 40 polygonal rooftop hypotheses. Most of the hypothesized roofs are rectangular, but six are L-shaped. Outlines of the extracted rooftops are shown in Figure 15. Alphabetic labels key into the discussion below.

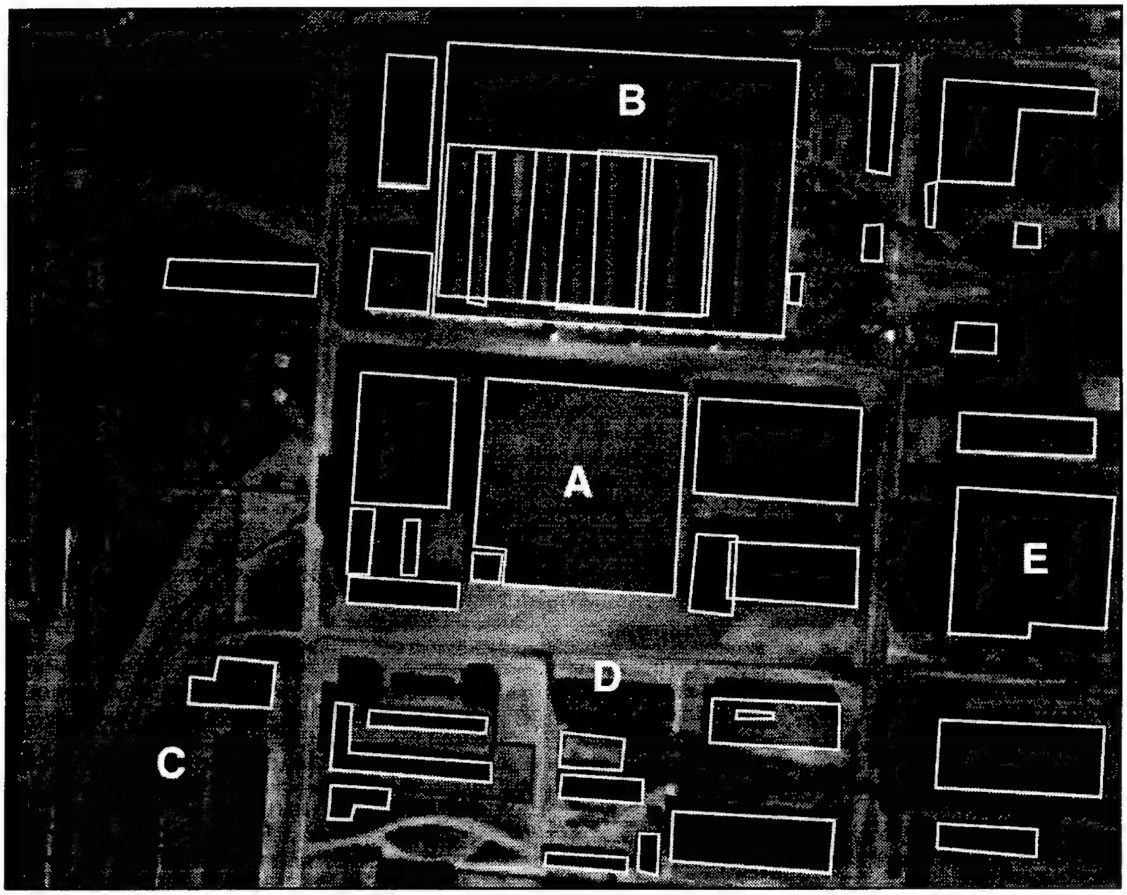


Figure 15: Roof hypotheses extracted from J3. Alphabetic labels are referred to in the text.

First, note that the overall performance is quite good for buildings entirely in view. Most of the major roof boundaries in the scene have been extracted, and in the central cluster of buildings (see area **A** in Figure 15), the segmentation is nearly perfect.

There were some false positives – polygons extracted that do not in fact delineate the boundaries of a roof. The most obvious example is the set of overlapping polygonal rooftops detected over the large building with many parallel roof vents (marked **B** in Figure 15). Note that the correct outer outline of this building roof is detected, however. The set of parallel roof vents on this building, coupled with the close proximity of other buildings and three tall smokestacks (and their shadows!) that occlude and fragment the building boundary in many of the images, make this one of the most challenging buildings in the site for rooftop detection, epipolar matching and intensity mapping.

There are also some false negatives, which are buildings that should have been detected, but weren't. The most prevalent example of this is a set of buildings (see **C**) that are only partially in view at the edge of the image. The current system is built implicitly around the idea of detecting complete building models; partial building structure information that is extracted is not carried along. Although the subsequent epipolar feature matching and multi-image line triangulation routines are already able to handle such building "fragments," additional code would be necessary to merge the partial building wireframes produced from different images into a single building model.

Label **D** marks a false negative that is in full view. Two adjacent corners in the rooftop polygon were missed by the corner extraction algorithm. Although a top-down virtual feature hypothesis can be invoked to insert a single missing corner in an incomplete rooftop polygon, there is no recovery mechanism when two adjacent corners are missing. It should be stressed that even though a single image was used here for bottom-up hypotheses, buildings that are not extracted in one image will often be found easily in other images with different viewpoints and sun angles.

There are several cases that cannot be strictly classified as false positives or false negatives. Several split-level buildings appearing along the right edge of the image (e.g. **E**) are outlined with single polygons rather than with one polygon per roof level. Some peaked roof buildings were also outlined, even though they do not conform to the generic assumptions underlying the system.

The results on this image illustrate an important fact - - the current rooftop detection algorithm is implicitly built around the goal of detecting complete building outlines. Partial building structure information (say from buildings that extend beyond the boundaries of the image) is not currently stored. Although subsequent routines for epipolar feature matching and multi-image line triangulation are already able to handle such building "fragments," additional code would be necessary to represent partial building wireframes in the final site model and to merge partial wireframes detected from different images into a single, coherent building model.

4.5 Multi-Image Epipolar Matching

After detecting a potential rooftop in one image, corroborating geometric evidence is sought in other images via epipolar polygon matching. As described in Section 3.3, potential matches for the edges of the rooftop polygon across multiple images are stored as votes in a building height histogram. After all votes have been tallied, the histogram bucket containing the most votes yields an estimate of the roof height in the scene and a set of correspondences between rooftop edges and image line segments from multiple views.

For the Model Board 1 experiment, the minimum and maximum values for the epipolar height histogram were chosen based on the range of Z-coordinates present in the set of measured ground truth points. The histogram contained 24 buckets with a height range of roughly 12 feet per bucket. After epipolar voting was completed for a rooftop polygon, correspondences were extracted from the histogram bucket containing the highest number of votes and those buckets immediately adjacent to it.

Epipolar matching of a rooftop hypothesis is considered to have failed when, for any edge in the rooftop polygon, no line segment correspondences are found in any image. This criterion was chosen because the 3D line triangulation algorithm will fail to converge in this case. Based on this criterion, epipolar matching failed on eight rooftop polygons. Six were either peaked or multi-layer roofs that did not fit the generic flat-roofed building assumption, and the other two were building fragments with some sides shorter than the minimum length threshold on the line segment data.

At this stage, we also removed six obviously incorrect building hypotheses by hand. Five of them comprised the set of overlapping polygons within the building labeled **B** in Figure 15. The sixth was the fenced-in area appearing directly below label **D** in that image. We believe that pointing to building hypotheses that are presented by the system to either accept or reject them is an acceptable level of interaction when creating a new site model. However, we are actively investigating methods for detecting and removing such mistakes automatically.

4.6 Multi-image Line Triangulation

Following epipolar rooftop verifcation and matching, a rigorous multi-image triangulation routine is performed to determine the precise size, shape, and position of the building roof in the local 3D site coordinate system. Outlines of the final set of triangulated rooftops are shown in Figure 16. The rightmost polygon in the image is noticeably incorrect. This polygon actually corresponds to a split-level building containing two roofs at different heights in the scene. Most of these split-level buildings were automatically filtered out during epipolar matching, but this one managed to survive. Determining how to automatically detect and remove such errors is an ongoing research issue – there is information contained in the epipolar histograms and triangulation residuals that has yet to be taken advantage of.

To evaluate the 3D accuracy of the triangulated building polygons, 21 roof vertices were identified where ground truth measurements are known. These locations are labeled in Figure 16 with numeric indices that are keyed to the file of Model Board 1 ground truth measurements. Table 2 shows the Euclidean distances between triangulated polygon vertices and their ground truth locations The average distance is 4.31 feet, which is reasonable given the level of geometric distortion present in the images (see Section 4.2).

Table 2: Euclidean distance (in feet) between triangulated and ground truth building vertex positions. Numeric indices correspond to the labeled positions in Figure 16.

index	error	index	error	index	error
10	6.21	**41**	3.53	**69**	2.78
17	1.20	**42**	5.21	**70**	2.12
18	13.70	**43**	4.70	**71**	2.62
22	3.41	**47**	3.88	**74**	2.62
37	6.75	**49**	4.22	**75**	4.87
39	3.59	**67**	3.85	**79**	2.30
40	4.30	**68**	4.18	**90**	4.58

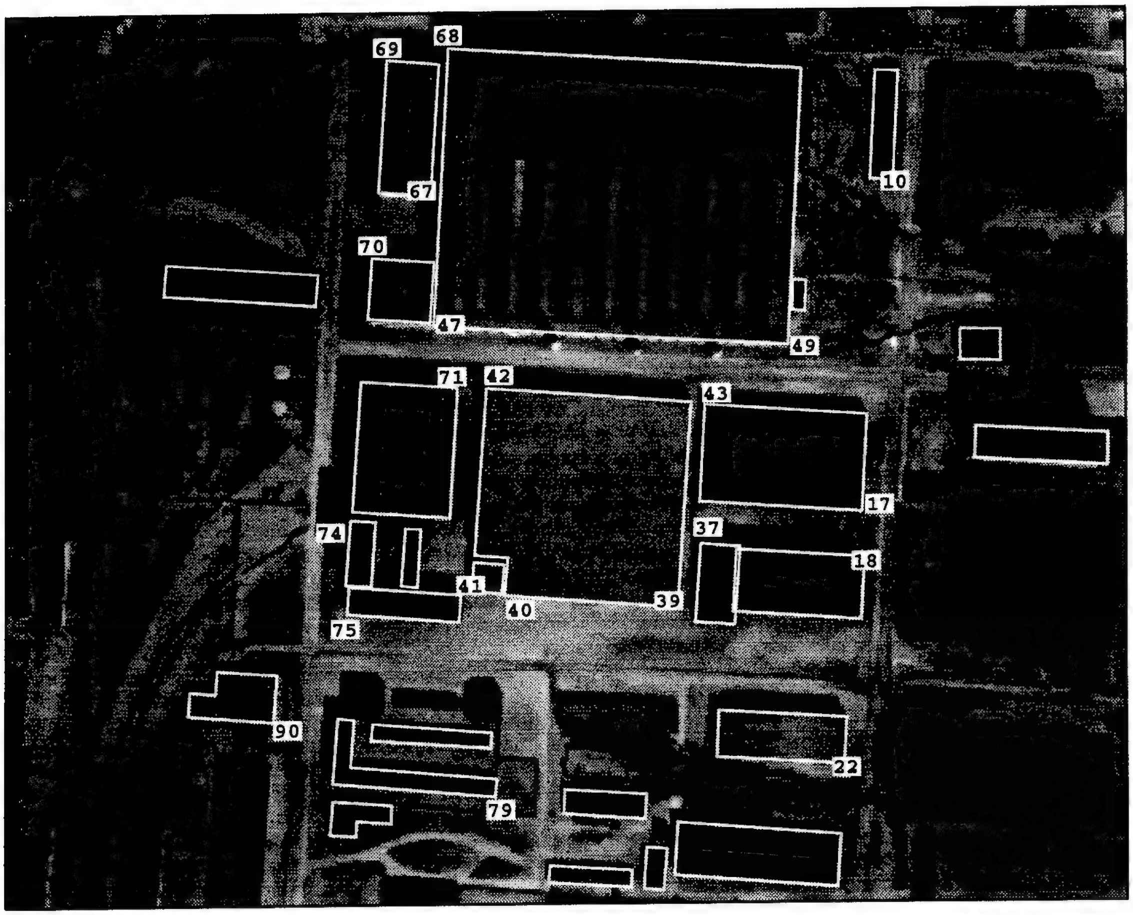


Figure 16: Verified and triangulated rooftops, projected back into image J3 (compare with Figure 15). Numeric labels mark 21 roof vertices where ground truth measurements are known.

It is instructive to decompose the distance error into its horizontal and vertical components. The average horizontal distance error is 3.76 feet, while the average vertical error is only 1.61 feet. This is understandable, since all observed rooftop lines are considered simultaneously when estimating the building height (vertical position), whereas the horizontal position of a rooftop vertex is primarily affected only by its two adjacent edges.

Also note that the error associated with point 18 appears to be an outlier – it is twice as large as the next largest distance. The building was not triangulated well, due in part to its extremely close proximity to a neighboring building, which interferes with correct matching and triangulation. It is no coincidence that the vertex error computed for the neighboring building is the second largest error.

After triangulation, each 3D rooftop polygon was extruded down to the ground to form a volumetric model. For the Model Board 1 site, we represented the ground as a horizontal plane with a Z-coordinate value determined from the ground truth measurements. More generally, we will soon be combining our symbolic building extraction routines with digital elevation maps produced by a terrain reconstruction system developed recently at UMass.

4.7 Projective Intensity Mapping

Projective mapping of image intensities (rendering) onto polygonal building model faces enhances their visual realism and provides a convenient storage mechanism for later symbolic extraction of detailed surface structure. For each of the 25 volumetric building models that were acquired, a surface intensity map was generated for each planar facet by projectively mapping intensity values from the images in which the facet is visible, and combining them using the "best piece" representation (BPR) method described in Section 3.5.

Figure 17 shows an example of the best piece represention and its construction for one particular building facet in the model. Figure 17(a) shows the set of sampled surface intensity maps for this facet, along with their associated shadow and occlusion labelings. This rectangular wall facet appears only in site images J1, J2, J6, and J8, and thus only these four images are sampled. In site image J1, part of the wall is cut by the image border, and is marked in the labeling image for the version from J1. Facet versions from J6 and J8 look darker because they are self-shadowed, i.e. oriented away from the light source. In site image J2 and J6, this wall is viewed from such an oblique angle that the textures mapped from these two images provide very little additional information over much of the wall surface. However, near the lower left of the wall there is another small building that occludes the wall in versions J1 and J8, but not in J2 and J6 due to the extreme obliquity of the viewing angle. This example illustrates why multiple images are necessary to see all the portions of this particular building face.

Figure 17(b) shows the BPR image synthesized using the facet images in Figure 17(a). The sunlit piece in J1 version is chosen as the exemplar piece for histogram adjustment of intensity values from different images. We can see that some regions in the BPR image contain the intensities from version J1, some others from J8 and J2. The intensities from different sources are consistent as well.

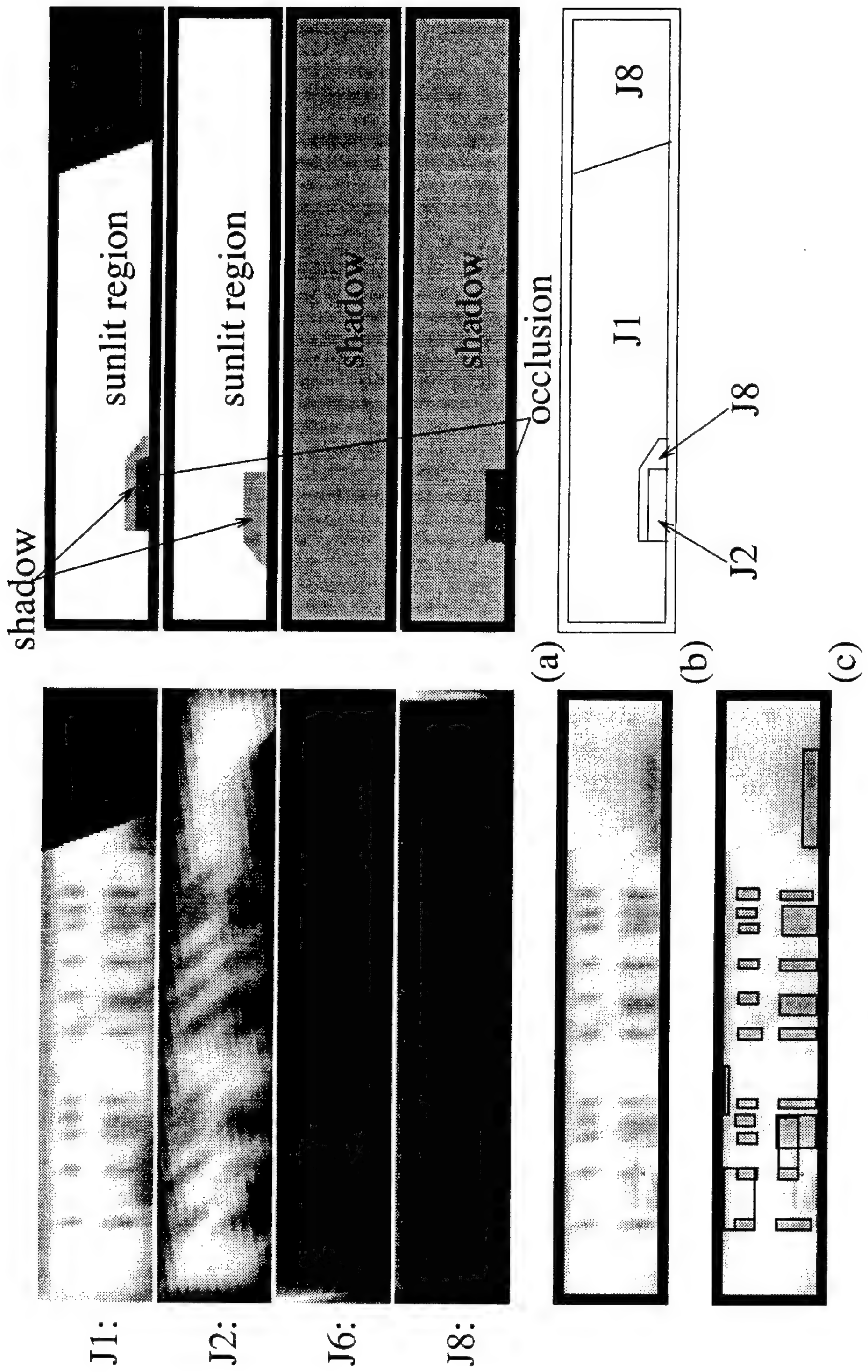


Figure 17: Construction and use of the best piece representation (BPR) of a building facet. (a) Sampled surface intensity maps and their label images. (b) BPR image of the facet, and its intensity sources. (c) The result of dark rectangle extraction on the BPR image.

By storing surface intensity information with the object, intensity mapping provides a convenient storage method for later symbolic extraction of detailed surface structures like windows, doors and roof vents. As one example, we have developed a generic algorithm for detecting dark, oriented rectangular patterns, as a method for extraction windows and doors on wall surfaces. Figure 17(c) shows the results of applying this algorithm to the BPR image in Figure 17(b). Twenty dark regions in Figure 17(c) are correctly extracted as hypothesized windows and doors and written into a new, refined model. There are also four false detections, which could potentially be removed automatically using knowledge-based constraints regarding the size and shape (range of aspect ratios) of true windows and doors. An interesting result in Figure 17(c) is that the detected rectangle on the right bottom of the image crosses over two regions whose intensities come from different image versions, showing that the BPR algorithm maintains good consistency of image intensities from different sources.

The purpose of model visualization is to provide a visually realistic rendering of the scene. A sample 3D site rendering is shown in Figure 18. The surfaces are texture-mapped using the BPR facet representatives constructed as described earlier. For example, the BPR synthesized wall facet from Figure 17 can be seen prominently in the image, when generated from this viewpoint. A second site rendering, from a viewpoint at the other end of the site, is shown in Figure 19. A sequence of rendered views constructed in this way has been used to construct a simulated video "fly-through" of the site, in order to demonstrate the level of realism achievable by these modeling techniques, and to investigate the use of visualization techniques for interactive evaluation of modeling results.

4.8 Summary

An evaluation of the IU algorithms that comprise the UMass automated building extraction system was performed using a sample site model acquisition task. The algorithms currently assume a generic class of flat roofed, rectilinear buildings. When run on image J3 of the Model Board 1 imagery, an automated building roof detector produced 40 rooftop hypotheses. Supporting evidence was located in other images via epipolar line segment matching, and the precise 3D shape and location of each building was determined by constrained multi-image line triangulation. Through a process of filtering and attrition, we ended up with 25 building models that represent most of the central buildings in the site. Projective mapping of intensity information from the images onto these polyhedral models results in a compelling site model display that can be used for rendering the scene from new views and for generating animated, video fly-throughs.

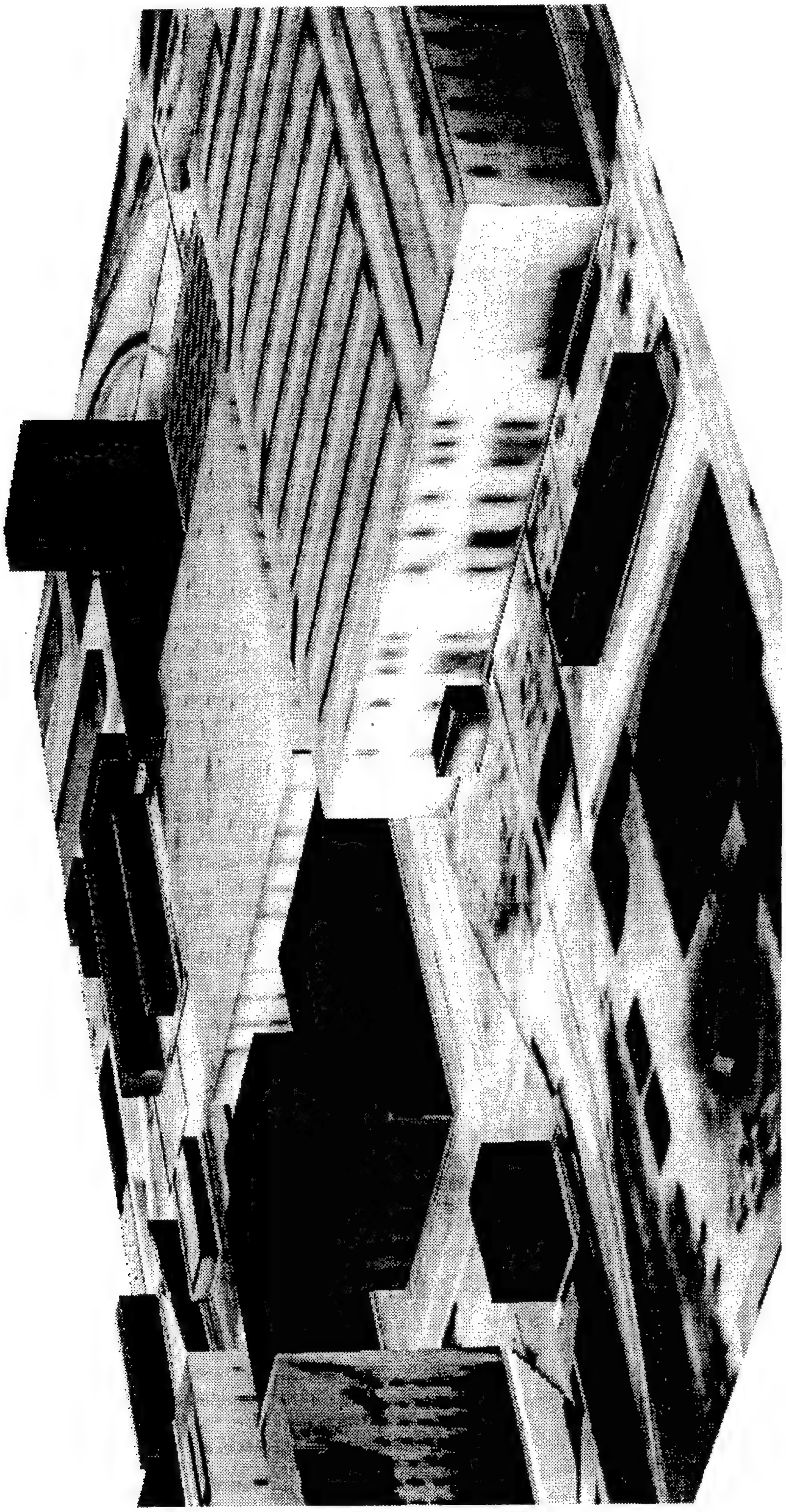


Figure 18: A rendered view of the full, intensity-mapped site model.

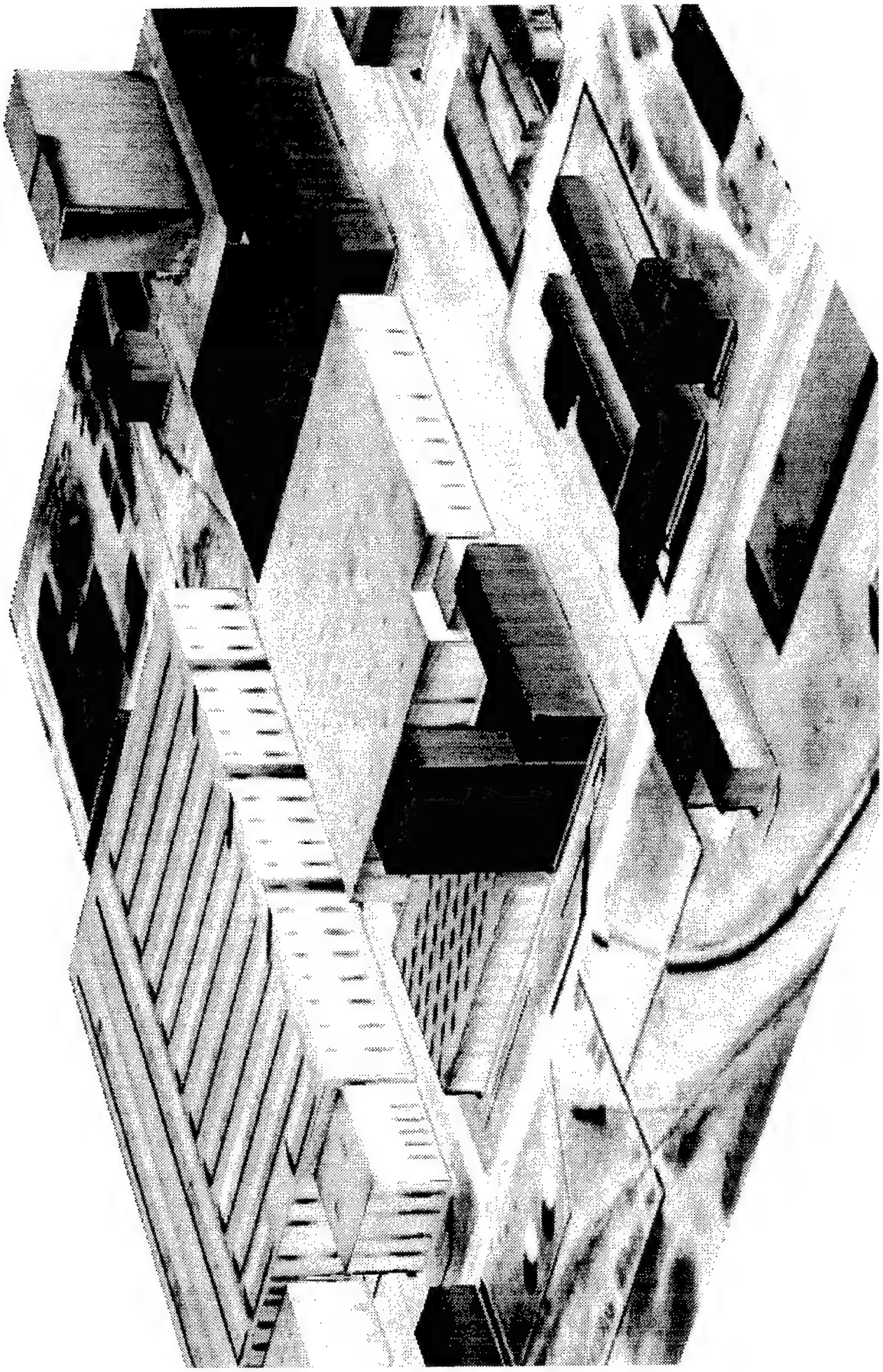


Figure 19: Another rendered view of the intensity-mapped site model.

5 Site Model Extension

The goal of site model extension is to find unmodeled buildings in new images and add them into an existing site model database. The main difference between model extension and model acquisition is that the prior site model can be used to make the building extraction process more efficient and more accurate. For example, areas already known to contain modeled buildings can be masked out in the new image, so that the computation time of the building extraction procedures can be devoted to unmodeled areas. Even if it is desired to verify that modeled buildings are still present in the new view, model-based prediction and verification procedures can be applied that are much more efficient than reacquiring the building models from scratch.

A second important aspect of site model extension is that the existing geometric site model can be used to refine the camera resection parameters provided by the Model-Supported Positioning (MSP) project. The nonlinearities of classified sensor imagery make it hard to resect, and even with state-of-the-art mathematical models of the sensor geometry, local corrections to the set of globally resected camera parameters may be necessary. For this purpose, we propose to use an automated model-to-image registration process that can determine local, model-based corrections that result in more accurate 3D triangulation of new structures.

In this section, we first briefly describe our model-to-image registration alorithms for locally refining the camera resection parameters provided with incoming imagery. This is followed by an example of model extension using the partial site model constructed in the last section.

5.1 Model-To-Image Registration

Our approach to model-to-image registration involves two components: 1) *model matching* to determine correspondences between model features and image features, and 2) *pose determination* to determine the precise geometric relationship between the image and the scene.

5.1.1 Model Matching

The goal of **model matching** is to find the correspondence between 3D features in a site model and 2D features that have been extracted from an image; in this case determining correspondences between edges in a 3D building wireframe and 2D extracted line segments from the image. The model matching algorithm described in [2] is being used. Based on a *local search* approach to combinatorial optimization, this algorithm searches the discrete space of correspondence mappings between model and image features for one that minimizes a match error function. The match error depends upon how well the projected model geometrically aligns with the data, as well as how much of the model is accounted for by the data. The result of model matching is a set of correspondences between model edges and image line segments, and an estimate of the transformation that brings the projected model into the best possible geometric alignment with the underlying image data.

The efficiency and success rate of the model-matching algorithm depends on finding a good initial set of potential model-to-image correspondences. This in turn depends on the quality and completeness of initial estimates of the image acquisition parameters. As a general rule-of-thumb, the difficulty of finding correspondences is directly proportional to the number of unknown acquisition parameters (unconstrained degrees of freedom in the model-to-image transformation space) and the amount of uncertainty in the acquisition parameters that *are* known. We currently assume the intrinsic (lens) parameters of the camera are accurately known, and that enough information is available to piece together a reasonable estimate of the extrinsic (pose) parameters [4]. In an operational setting, an initial estimate of camera pose for each local area of interest can be derived from the resection parameters provided by the MSP project.

5.1.2 Pose Determination

The second aspect of model-to-image registration is precise **pose determination**. It is important to note that since model-to-image correspondences are being found automatically, the pose determination routine needs to take into account the possibility of mistakes or *outliers* in the set of correspondences found. The robust pose estimation procedure described in [7] is being used.

At the heart of this code is an iterative, weighted least-squares algorithm for computing pose from a set of correspondences that are assumed to be free from outliers. The pose parameters are found by minimizing an objective function that measures how closely projected model features fall to their corresponding image features. Although similar objective functions have been proposed elsewhere, two novel solution techniques distinguish this algorithm from past approaches. First, both rotation and translation parameters are solved for simultaneously, which makes more effective use of the geometric constraints and is more accurate in the presence of noise than techniques that decompose the problem by solving for rotation first, followed by translation. Second, the nonlinear least-squares optimization algorithm used to solve for rotation and translation is based on the quaternion representation of rotations, which provides much better convergence properties than solution methods based on Euler angles.

It is well known that least-squares optimization techniques can fail catastrophically when outliers are present in the data. For this reason, the basic pose algorithm described above is embedded inside a least-median-squares (LMS) procedure that repeatedly samples subsets of correspondences to find one devoid of outliers. This approach is called least median squares because it in effect minimizes the median-squared residual distance error rather than the mean-squared distance. LMS is robust over data sets containing up to 50% outliers. The final results of pose determination are a set of camera pose parameters and a covariance matrix that estimates the accuracy of the solution.

5.2 Model Extension Example

The model extension process involves registering a current geometric site model with a new, incoming image, and then focusing on unmodeled areas to try to recover new buildings that have been recently built, that were previously unseen, or that for some other reason are not present in the site model database. We illustrate this process using the partial site model constructed in Section 4, and image J8 from the Radius Model Board 1 dataset. Although this is not strictly speaking a new image, since it was one of the eight images used during initial model acquisition, it will suffice for this example since we did not find all the buildings during that previous model construction phase.

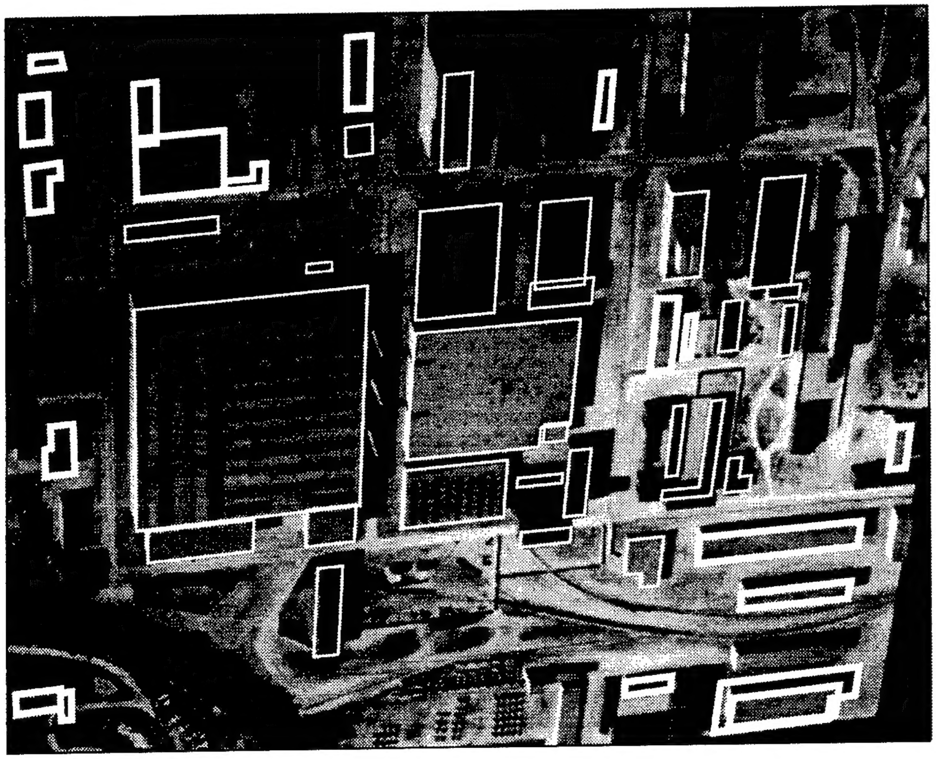


Figure 20: Previous site model rooftops (thin lines) projected onto image J8, via model-to-image registration. New rooftop hypotheses produced during model extension are also shown (thick lines).

Results of registering image J8 with the partial site model can be seen in Figure 20, which shows projected building rooftops from the site model overlaid on the image (thin lines). Based on this model-to-image registration, the areas containing buildings already in the site model were masked off, and the building rooftop detector was run on the unmodeled areas in the image. A set of 19 rooftop hypotheses was generated, also shown in Figure 20 (thick lines). Once again, the detector did a reasonably good job at delineating the significant building rooftops lying completely within the image. Only one hypothesis is notably incorrect - - in the lower right hand corner a set of building shadows have been incorperated into a spurious rooftop hypothesis surrounding an actual roof, which *was* detected correctly.

Based on the new set of rooftop hypotheses, the multi-image epipolar matching and

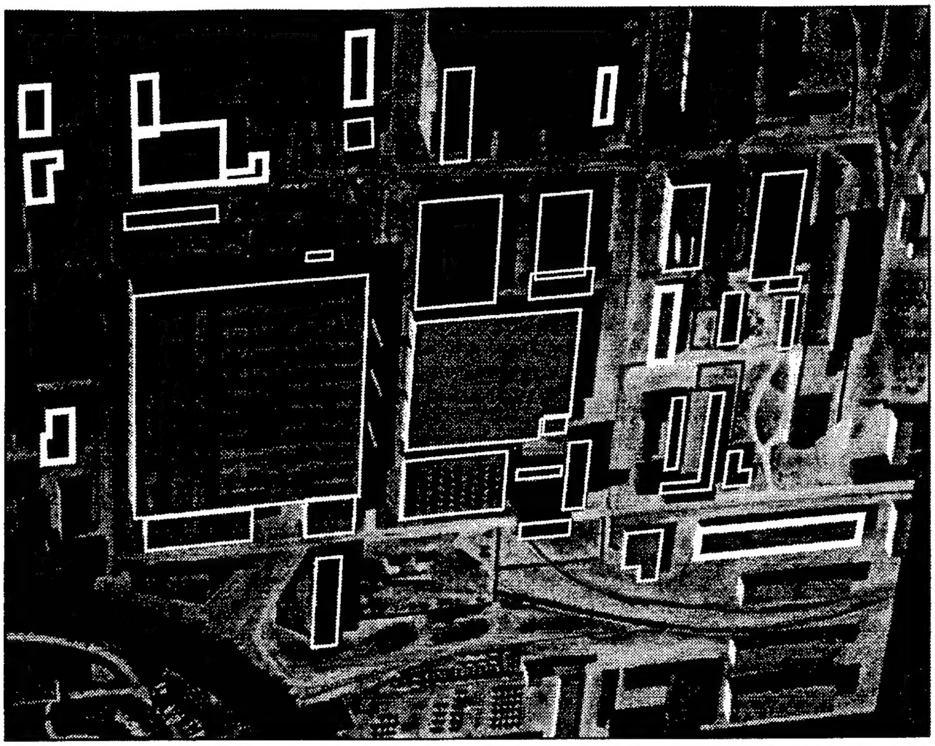


Figure 21: Updated site model projected onto image J8. Thick lines delineate buildings that were added based on model extension, thin lines show the previous site model.

constrained multi-image triangulation procedures from Section 4 were applied, again using images J1–J8, to verify the new buildings and construct 3D volumetric building models. Only 10 hypotheses survived the verification and triangulation process. These were added to the site model database, to produce the extended model shown in Figure 21. The main reason for failure among building hypotheses that were not verified was that they represent buildings located at the periphery of the site, in an area which is not visible in very many of the eight views. If more images were used with greater site coverage, we expect that more of these hypotheses would survive the multi-image verification process.

6 Summary and Future Work

A set of IU algorithms for automated site model acquisition and extension are being developed at the University of Massachusetts. The UMass design philosophy emphasizes model-directed processing, rigorous 3D perspective camera equations, and fusion of information across multiple images for increased accuracy and reliability. This year, a system for automated building extraction was developed. The building extraction system currently assumes a generic class of flat-roofed, rectilinear buildings. Other required input parameters to the system include a set of images of the site, a user-define local site model coordinate system, camera resection parameters relating each image the to site model coordinate system, and a digital terrain map to place the extracted buildings upon.

To acquire a new site model, an automated building detector is run on one image to hypothesize potential building rooftops. Supporting evidence is located in other images via epipolar line segment matching, and the precise 3D shape and location of each building is determine by multi-image triangulation. Projective mapping of image intensity information onto these polyhedral building models results in a realistic site model that can be rendered using virtual "fly-through" graphics. An experimental case study of the model acquisition process was carried out using images J1–J8 of the RADIUS Model Board 1 data set. In this experiment, 25 building models were generated, covering a large portion of the model board site.

To perform model extension, a prior site model is registered to new images, and model acquisition procedures are focused on previously unmodeled areas. In an operational scenario, this process would be repeated as new images become available, gradually accumulating evidence over time to make the site model database more complete and more accurate. An example of extending the partial Model Board 1 site model generated during model acquisition was presented. Model-to-image registration techniques were also presented that can be used to automatically determine model-based, local corrections to the resected camera parameters provided with each image.

Several avenues for system improvement are open. One high priority is to add capabilities for detecting and triangulating peaked roof buildings. Another significant improvement would be extending the epipolar matching and triangulation portions of the system to analyze why a particular building roof hypothesis failed to be verified. There are many cases where the rooftop detector has outlined split-level buildings with a single roof polygon. This currently causes the subsequent epipolar verification procedure to fail, since all lines in the polygon are assumed to be at the same height. However, a careful analysis of the height histogram in these cases reveals it to be bimodal, meaning that some lines have been found to be at one height, while some occur at another. Automatic detection of these situations, followed by splitting the rooftop hypothesis into two separate hypotheses, one for each roof level, would result in an improvement in system performance.

Our symbolic building extraction procedures will soon be combined with a correlation-based terrain extraction system developed at UMass. The two techniques clearly complement each other: the terrain extraction system can determine a digital elevation map upon which the volumetric building models rest, and the symbolic building extraction procedures can identify building occlusion boundaries where correlation-based terrain recovery is expected to behave poorly. A tighter coupling of the two systems, where an initial digital elevation map is used to focus attention on distinctive humps that may be buildings, or where correlation-based reconstruction techniques are applied to building rooftop regions to identify fine surface structure like roof vents and air conditioner units, may also be investigated.

Acknowledgements

This project would not be possible without the hard work and dedication of the Radius team members: Yong-Qing Cheng, Chris Jaynes, Frank Stolle, and Xiaoguang "XG" Wang. We would also like to acknowledge the technical and administrative support of Robert Heller, Jonathan Lim, Janet Turnbull, Laurie Downey and Fred Weiss.

References

[1] American Society of Photogrammetry, *Manual of Photogrammetry,* Fourth Edition, American Society of Photogrammetry, Falls Church, VA, 1980.

[2] J.R. Beveridge and E. Riseman, "Hybrid Weak-Perspective and Full-Perspective Matching," *Proc. Computer Vision and Pattern Recognition,* Champaign, IL, 1992, pp. 432-438.

[3] M. Boldt, R. Weiss and E. Riseman, "Token-Based Extraction of Straight Lines," *IEEE Transactions on Systems, Man and Cybernetics,* Vol. 19, No. 6, 1989, pp. 1581–1594.

[4] R. Collins, A. Hanson, E. Riseman and Y. Cheng, "Model Matching and Extension for Automated 3D Site Modeling," *Proceedings Arpa Image Understanding Workshop,* Washington, DC, April 1993, pp. 197–203.

[5] A. Huertas, C. Lin and R. Nevatia, "Detection of Buildings from Monocular Views of Aerial Scenes using Perceptual Grouping and Shadows," *Proc. Arpa Image Understanding Workshop,* Washington, DC, April 1993, pp. 253–260.

[6] C. Jaynes, F. Stolle and R. Collins, "Task Driven Perceptual Organization for Extraction of Rooftop Polygons," *Proceedings Arpa Image Understanding Workshop,* Monterey, CA, November 1994, pp. 359–365.

[7] R. Kumar and A. Hanson, "Robust Methods for Estimating Pose and Sensitivity Analysis," *CVGIP: Image Understanding,* Vol. 60, No. 3, November 1994, pp. 313-342.

[8] J. Mundy, R. Welty, L. Quam, T. Strat, W. Bremner, M. Horwedel, D. Hackett and A. Hoogs, "The RADIUS Common Development Environment," *Proceedings of the Darpa Image Understanding Workshop,* San Diego, CA, January 1992, pp. 215–226.